In-Class Worksheet 11: Computable Reductions

Clarissa Littler

November 22, 2014

General notes on the solutions here: try to prove to yourself that the reductions defined are, in fact, reductions by testing to make sure that for any reduction *from* A *to* B that a string is in A if and only if its image under the reduction is in B.

Construct the following reductions

- A computational reduction from the ordinary halting problem to $H'_{TM} = \{ M \mid M \text{ halts on every input}\}$ Answer: We define this reduction as follows

```
R := On input (M,w)
  1. Define TM M' := On input y:
                       1. Run M(y)
                       2. if M(y) then accept else loop
  2. Write M' to tape
```

- A computational reduction *from* the language $\{ 0^{2^n} \mid n \geq 0\}$ to the language $\{0\}$ (this is actually an example of a more general principle) Answer:

```
R := On input w:
  1. Run decider D(w) where D is the decider for the powers of 2 language
  2. If D(w) Then write 0 to tape Else write 1 to tape
```

- A reduction from the language $\{(N,w) \mid N \text{ is an NFA that accepts } w\}$ *to* the language $\{(M,w) \mid M \text{ is a DFA that accepts } w\}$

```
R := On input (N,w):
  1. Run NFA -> DFA conversion on N to produce D
  2. Write (D,w) to tape
```

- A reduction from the $\mathrm{E_{TM}} = \{\ \mathrm{M} \mid \mathrm{L(M)} = \emptyset\}$ to $\mathrm{L_{TM}} = \{\ \mathrm{M} \mid \mathrm{M}$ loops on every input$\}$

  ```
  R := On input M:
    1. Define TM M' := On input w:
                         1. Run M(w)
                         2. If M(w) Then accept Else loop
    2. Write M' to tape
  ```

- Construct a reduction from $\overline{A_{TM}}$ to $\mathrm{E_{TM}}$

  ```
  R := On input (M,w):
    1. Define TM M' := On input y:
                         1. Run M(w)
                         2. If M(w) Then accept Else Reject
    2. Write M' to tape
  ```

- Prove that $A_{TM} \leq \overline{E_{TM}}$ We can actually use the same reduction above, which is the reason why, in general, $A \leq_m B$ implies $\overline{A} \leq_m \overline{B}$

- Prove that $0^n 1^n \leq 0^n$ This is basically the same trick as above

  ```
  R := On input w:
    1. Run PDA corresponding to {0^n1^n} on w
    2. If it accepts, write 0 to tape, otherwise write 1 to tape
  ```

- Prove that $\mathrm{EQ_{TM}}$ is not decidable The idea here is that we can show that $\mathrm{EQ_{TM}}$ is not decidable by reducing a language that isn't decidable *to* it, by which we're going to reduce $\mathrm{E_{TM}}$ to $\mathrm{EQ_{TM}}$.

  ```
  R := On input M;
    1. Define TM M' := On input w:
                         1. reject
    2. Write (M,M') to tape
  ```

  in other words, we reduce $\mathrm{E_{TM}}$ to $\mathrm{EQ_{TM}}$ by comparing our candidate member of E_{TM} against a language we *know* rejects all inputs.